Hybrid Forecasting Models for Drug Inventory Prediction in Hospital Pharmacies

Yuxin Fan* and Siye Wu†

*School of Engineering and Applied Science, University of Pennsylvania, Canada, Toronto
Email: yuxinfan@alumni.upenn.edu
†Simon Business School, University of Rochester, Canada, Toronto
Email: april.siyewu@hotmail.com

Abstract—Accurate and efficient drug inventory management is crucial for hospital pharmacies to avoid overstocking, minimize wastage, and ensure continuous patient care. This study proposes a hybrid forecasting framework integrating XGBoost, Prophet, and SARIMAX to improve monthly consumption predictions at the drug-manufacturer level. Through rolling-window forecasting and advanced feature engineering, the proposed approach addresses challenges such as seasonality, trend shifts, and sparse data. Experimental results demonstrate significant improvements in prediction accuracy and robustness across diverse drug consumption scenarios.

Index Terms—Drug Inventory, Forecasting Models, XGBoost, Prophet, SARIMAX

I. INTRODUCTION

Demand forecasting plays a critical role in the pharmaceutical industry, impacting inventory management, production planning, and supply chain optimization. Accurate demand predictions enable pharmaceutical companies to ensure drug availability while minimizing costs associated with overstocking or stockouts. Several techniques have been proposed to address these challenges, ranging from traditional statistical models to machine learning approaches.

For instance, Taylor and Letham [1] introduced scalable forecasting methods for large-scale applications, providing a robust foundation for predictive modeling in healthcare. Hyndman and Athanasopoulos [2] explored advanced time-series forecasting techniques, demonstrating their applicability in dynamic environments. Chen and Guestrin [3] proposed the XGBoost framework, a powerful tool for regression and classification tasks, which has been successfully adapted to demand forecasting scenarios. Meng et al. [4] compared the performance of traditional and machine learning methods, emphasizing the advantages of integrating LSTM models for improved accuracy. Xu et al. [5] demonstrated the effectiveness of hybrid models combining linear regression and deep learning techniques for time-series predictions.

In addition to these foundational works, recent studies have highlighted the potential of integrating domain-specific factors and hybrid approaches for pharmaceutical demand forecasting. Poyraz et al. [6] applied machine learning algorithms to predict pharmacy-level drug sales, achieving notable accuracy through models such as Random Forest and decision trees. Beyca et al. [7] further integrated time-series and regression models, addressing external influences like promotions and pricing strategies. Haider [8] emphasized the importance of external data sources, such as market trends and regulatory changes, in enhancing forecasting accuracy. Rathipriya et al. [9] explored hybrid neural network models, validating their robustness in handling complex, nonlinear patterns.

Other research has focused on developing machine learning models for specific use cases. Mbonyinshuti et al. [10] applied Random Forest techniques to predict essential medicine consumption in Rwanda, demonstrating the utility of data-driven approaches for public health. Ghousi et al. [11] investigated data mining techniques, revealing how association rules and artificial neural networks can uncover hidden patterns in drug consumption data. Siddiqui et al. [12] proposed a hybrid ARIMA-Holt's Winter model, achieving superior accuracy for pharmaceutical demand forecasting by leveraging the strengths of individual algorithms.

Building on these foundational studies, we propose a novel hybrid forecasting framework that integrates XGBoost, SARIMAX, and Prophet. Our approach is designed to overcome the limitations of individual models by:

- Leveraging XGBoost's strength in modeling non-linear relationships and feature interactions.
- Utilizing SARIMAX to explicitly incorporate external variables and capture seasonality.
- Exploiting Prophet's ability to decompose trends and handle missing data robustly.

The proposed framework aims to provide a scalable and adaptable solution tailored to the unique challenges of drug inventory forecasting. By combining these complementary methodologies, we address key gaps in existing research and offer a comprehensive approach to improve prediction accuracy and robustness.

The remainder of this paper is organized as follows: Section II details the methodology, including data preprocessing and model design. Section III presents experimental results and discussions, highlighting the advantages of the proposed framework. Finally, Section IV concludes the study and outlines future research directions.

II. METHODOLOGY

A. Data Processing

The raw dataset consists of monthly drug consumption records, including drug name, manufacturer, and associated features such as monthly consumption values, inventory amounts, and proportions. To ensure data quality and consistency, a comprehensive preprocessing pipeline was applied. Initially, duplicate entries were removed by identifying and merging records with the same combination of drug name, manufacturer, and date. Missing values in critical features were handled using interpolation for time series variables, while records with excessive missing data were excluded entirely. To facilitate downstream analysis and ensure comparability, continuous variables such as consumption and inventory values were standardized where appropriate.

Outlier detection was conducted to address extreme values in the consumption data, leveraging a dynamic rolling-window approach. Specifically, for each group of drug-manufacturer combinations, rolling statistics including mean, standard deviation, and quantile thresholds (5th and 95th percentiles) were computed using a sliding window of seven months. Potential outliers were identified based on their Z-scores, with values exceeding a threshold of 3 flagged as anomalies, as well as those falling outside the computed quantile range. Detected outliers were adjusted to the nearest valid boundary to preserve the continuity of the time series while mitigating the influence of extreme values.

To ensure the dataset included only high-quality samples for modeling, a rigorous sample selection process was employed. Groups with fewer than six months of non-zero consumption data or those exhibiting extreme sparsity, defined by a low proportion of non-zero values, were excluded. Additionally, the temporal autocorrelation of the consumption data was assessed using the autocorrelation function (ACF), and groups with insufficient autocorrelation were removed. Further filtering criteria included variance thresholds to exclude groups with overly stable data, limits on missing data ratios for derived features, and bounds on skewness to prevent heavily imbalanced target distributions. To ensure temporal relevance, only groups with at least one record on or after August 31, 2024, were retained. Moreover, feature correlation analysis was performed to ensure predictive relevance, with groups failing to meet minimum correlation thresholds excluded from the final dataset.

Finally, feature engineering was conducted to enhance the predictive power of the dataset. Derived features such as lagged consumption values (e.g., previous month's consumption), rolling statistics (e.g., mean, variance, and percent changes), and seasonality indicators encoded with trigonometric functions were created to capture temporal dependencies and periodic trends. This preprocessing pipeline ensured that the final dataset was robust, informative, and well-suited for downstream predictive modeling tasks. Details of the sample selection algorithm are provided in Appendix A.

B. Models

The study evaluates three models—XGBoost, SARIMAX, and Prophet—both independently and within a hybrid framework. Each model contributes unique capabilities. XGBoost handles short-term predictions effectively, SARIMAX captures long-term trends, and Prophet excels in handling irregular seasonal patterns.

C. SARIMAX with Exogenous Variables

SARIMAX extends the traditional ARIMA model by incorporating external (exogenous) variables, denoted as X_t. The SARIMAX model is represented as:

$$y_t = \phi(B)\theta(B)^{-1}\left(c + \mathbf{X}_t\beta + \epsilon_t\right), \tag{1}$$

where:
- y_t: The target variable (e.g., monthly drug consumption).
- $\phi(B)$: The autoregressive (AR) operator.
- $\theta(B)$: The moving average (MA) operator.
- c: A constant term.
- $\mathbf{X}_t$: A vector of exogenous variables at time t.
- β: The coefficient vector for $\mathbf{X}_t$.
- ϵ_i: White noise error term.

1) Exogenous Variables: To enhance prediction accuracy, the following exogenous variables are included:

1) **Lagged Values**:

$$\text{lag}_k = y_{t-k}, \quad k \in \{1, 3, 6, 12\}, \tag{2}$$

capturing the delayed impact of past consumption.

2) **Rolling Statistics**:

$$\text{Rolling Mean}_k = \frac{1}{k}\sum_{i=1}^{k} y_{t-i}, \tag{3}$$

$$\text{Rolling Std}_k = \sqrt{\frac{1}{k}\sum_{i=1}^{k}(y_{t-i} - \text{Mean})^2}, \tag{4}$$

representing the moving average and variability over a specified window.

3) **Exponential Weighted Moving Average (EWMA)**:

$$\text{EWMA}_\alpha = \alpha y_t + (1-\alpha) \cdot \text{EWMA}_{\alpha, t-1}, \tag{5}$$

emphasizing recent observations with a smoothing factor α.

4) **Seasonality Variables**: Monthly seasonality is encoded using trigonometric functions:

$$\text{Month\_sin} = \sin\left(\frac{2\pi \cdot \text{Month}}{12}\right), \tag{6}$$

$$\text{Month\_cos} = \cos\left(\frac{2\pi \cdot \text{Month}}{12}\right). \tag{7}$$

5) **Percentage Change**: Measuring relative changes over time:

$$\text{Pet\_Change}_1 = \frac{y_t - y_{t-1}}{y_{t-1}}, \tag{8}$$

$$\text{Pet\_Change}_3 = \frac{y_t - y_{t-3}}{y_{t-3}}. \tag{9}$$

6) **Trend and Volatility**:

$$\text{Trend Strength} = \frac{1}{k} \sum_{i=1}^{k} |y_{t-i} - y_{t-t-1}|, \qquad (10)$$

$$\text{Volatility} = \text{Rolling Std}_k. \qquad (11)$$

These variables capture both historical patterns and contextual dynamics, enabling SARIMAX to model complex dependencies.

D. XGBoost Model

XGBoost is a gradient boosting model that constructs an ensemble of decision trees to predict the target variable. It excels at capturing non-linear relationships and interactions in the data. For time-series forecasting, exogenous features and rolling-window predictions are incorporated.

The XGBoost model can be represented as:

$$\hat{y}_t = F(x_t) = \sum_{k=1}^{K} f_k(x_t), \quad f_k \in \mathcal{F}, \qquad (12)$$

where:

- $\hat{y}_t$: The predicted value at time t,
- x_t: The input features (e.g., lagged values, rolling mean, seasonality),
- f_k: The k-th decision tree,
- $\mathcal{F}$: The function space of decision trees.

1) Model Training: The XGBoost model is trained iteratively using a rolling-window approach. At each iteration, a grid search is performed over the following hyperparameter space:

- Number of estimators (n_estimators): [100, 200, 300],
- Learning rate (learning_rate): [0.01, 0.05, 0.1],
- Maximum tree depth (max_depth): [3, 5, 7],
- Subsample ratio (subsample): [0.8, 1.0],
- Column sampling (colsample_bytree): [0.8, 1.0].

E. Prophet Model

Prophet is a time-series forecasting model developed by Facebook that explicitly decomposes the data into trend, seasonality, and holiday components. It is particularly effective for handling missing values, outliers, and irregular patterns in the data.

The model can be represented as:

$$y_t = g(t) + s(t) + h(t) + \epsilon_t, \qquad (13)$$

where:

- $g(t)$: Trend component, representing long-term growth or decay,
- $s(t)$: Seasonal component, modeled using Fourier series,
- $h(t)$: Holiday effects or special events,
- ϵ_t: White noise error term.

1) Parameter Tuning: Prophet's key hyperparameters include:

- seasonality_mode: Additive or multiplicative,
- changepoint_prior_scale: Controls flexibility of the trend component,
- seasonality_prior_scale: Controls the weight of seasonal components.

A grid search over these parameters is performed to identify the optimal configuration for each drug-manufacturer combination.

F. Dynamic Feature Engineering

Feature engineering is crucial for enhancing model performance. Besides the exogenous variables described above, outlier detection is a key preprocessing step.

1) Outlier Detection and Handling: Outliers can distort predictions and degrade model performance. This study employs two approaches for outlier detection:

- **Z-score Method**: The Z-score for each data point y_i is calculated as:

$$Z_i = \frac{y_i - \mu}{\sigma}, \qquad (14)$$

  where μ is the mean and σ is the standard deviation of the dataset. Data points with $|Z_i| > 3$ are identified as outliers.

- **Interquartile Range (IQR) Method**: The IQR is defined as:

$$\text{IQR} = Q_3 - Q_1, \qquad (15)$$

  where Q_1 and Q_3 are the 25th and 75th percentiles, respectively. A data point y_i is considered an outlier if:

$$y_i < Q_1 - 1.5 \cdot \text{IQR} \quad \text{or} \quad y_i > Q_3 + 1.5 \cdot \text{IQR}. \qquad (16)$$

To mitigate the effect of outliers, values exceeding these thresholds are capped at the 5th and 95th percentiles of the data distribution:

$$y_i = \begin{cases} \text{Percentiles}, & \text{if } y_i < \text{Percentiles} \\ \text{Percentile}_{95}, & \text{if } y_i > \text{Percentile}_{95}. \end{cases} \qquad (17)$$

2) Rolling-Window Forecasting: To adapt to dynamic consumption patterns, a rolling-window mechanism is implemented. At each prediction step t, the model is trained using historical data $\{y_1, y_2, \ldots, y_t\}$, and the prediction for $t + 1$ is made. The window then updates to include the latest observation, ensuring that the model adapts to recent trends.

III. EXPERIMENTS

A. Experimental Setup

The dataset consists of monthly drug consumption records, including:

- Drug Name
- Manufacturer
- Monthly Consumption

Models are evaluated using the following metrics:

1) **Root Mean Squared Error (RMSE)**:

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}. \tag{18}$$

2) **Mean Absolute Error (MAE)**:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|. \tag{19}$$

3) **Symmetric Mean Absolute Percentage Error (SMAPE)**:

$$\text{SMAPE} = \frac{1}{n}\sum_{i=1}^{n}\frac{|y_i - \hat{y}_i|}{(|y_i| + |\hat{y}_i|)/2} \times 100\%. \tag{20}$$

B. Results

The performance of the models is summarized in Table I.

IV. Discussion and Future Work

A. Model Comparison and Hybrid Framework

While each model has unique strengths and limitations, integrating their complementary capabilities within a hybrid framework provides significant advantages:

- **XGBoost** handles nonlinear relationships effectively.
- **SARIMAX** captures seasonality and leverages external variables.
- **Prophet** provides robust decomposition of trend and seasonality.

The hybrid approach enables the framework to adapt to different data characteristics, achieving better overall prediction performance. For example, SARIMAX improves accuracy when seasonality dominates, whereas XGBoost captures complex interactions in data with dynamic patterns.

B. Future Work

To further improve drug inventory predictions, the following directions are proposed for future research:

- **Hybrid Framework Expansion**: Extend the hybrid framework by integrating additional models, such as LightGBM, Transformer-based architectures, or ensemble methods to capture both short-term fluctuations and long-term trends.
- **External Variable Enrichment**: Explore additional exogenous variables such as patient inflow, epidemic outbreak data, seasonal illnesses (e.g., flu seasons), or hospital-specific factors to improve prediction accuracy.
- **Automated Model Selection**: Implement automated hyperparameter tuning and model selection techniques (e.g., Bayesian optimization) to improve forecasting performance with minimal manual intervention.

By addressing these areas, the proposed framework can become a more robust and scalable solution for hospital pharmacy inventory management, ensuring continuous patient care and reducing operational inefficiencies.

V. Conclusion

This study evaluated the performance of XGBoost, SARIMAX, and Prophet for drug inventory prediction in hospital pharmacies. The main findings are as follows:

- XGBoost outperformed SARIMAX and Prophet in terms of RMSE and SMAPE for short-term predictions.
- SARIMAX demonstrated better performance when external (exogenous) variables were included and seasonal trends dominated the data.
- Prophet performed well in capturing long-term seasonality and trends but showed limitations in handling highly sparse data.

The hybrid framework, by combining these models, shows promise for improving prediction accuracy and robustness. Future work will focus on integrating real-time data streams and exploring additional hybrid models, such as Transformer-based architectures, to further enhance predictive capabilities.

VI. References

References

[1] S. J. Taylor and B. Letham, "Forecasting at scale," *The American Statistician*, vol. 72, no. 1, pp. 37–45, 2018.
[2] R. J. Hyndman and G. Athanasopoulos, *Forecasting: Principles and Practice*, Melbourne, Australia: OTexts, 2018.
[3] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 785–794.
[4] J. Meng et al., "Comparative analysis of Prophet and LSTM models in drug sales forecasting," *Journal of Physics: Conference Series*, vol. 1910, no. 1, p. 012059, 2021.
[5] W. Xu et al., "A hybrid modelling method for time series forecasting based on a linear regression model and deep learning," *Applied Intelligence*, vol. 49, no. 7, pp. 2875–2888, 2019.
[6] I. Poyraz and A. Gürhanlı, "Drug Demand Forecasting for Pharmacies with Machine Learning Algorithms," *International Journal of Engineering Research and Applications*, vol. 10, no. 7, pp. 51–54, 2020.
[7] S. İmece and Ö. F. Beyca, "Demand Forecasting with Integration of Time Series and Regression Models in Pharmaceutical Industry," *International Journal of Advanced Engineering Pure Science*, vol. 34, no. 3, pp. 415–425, 2022.
[8] N. N. Haider, "Enhancing Forecasting Accuracy in the Pharmaceutical Industry: A Comprehensive Review of Methods, Models, and Data Applications," *Journal of Scientific and Engineering Research*, vol. 9, no. 2, pp. 156–160, 2022.
[9] R. Rathipriya et al., "Demand Forecasting Model for Time-Series Pharmaceutical Data Using Neural Networks," *Neural Computing and Applications*, vol. 35, pp. 1945–1957, 2023.
[10] F. Mbonyinshuti et al., "The Prediction of Essential Medicines Demand: A Machine Learning Approach Using Consumption Data in Rwanda," *Processes*, vol. 10, no. 26, pp. 1–13, 2022.
[11] R. Ghousi et al., "Application of Data Mining Techniques in Drug Consumption Forecasting to Help Pharmaceutical Industry Production Planning," *Proceedings of the International Conference on Industrial Engineering and Operations Management*, pp. 1162–1165, 2012.
[12] R. Siddiqui et al., "A Hybrid Demand Forecasting Model for Greater Forecasting Accuracy: The Case of the Pharmaceutical Industry," *Supply Chain Forum: An International Journal*, vol. 22, no. 3, pp. 1–13, 2021.

Appendix

The following pseudocode outlines the sample selection process used to ensure the quality and relevance of the dataset for modeling tasks:

Algorithm 1 Sample Selection Algorithm

Require: Cleaned data df, configuration thresholds config
Ensure: Filtered dataset $final\_df$

1: $final\_df \leftarrow \emptyset$
2: **for** each unique combination of drug name and manufacturer (d, m) in df **do**
3:      $group\_data \leftarrow$ subset of df for (d, m)
4:      **if** length of $group\_data <$ config.min_months **or** sum of consumption $= 0$ **then**
5:          **Skip group**   ▷ Insufficient or sparse data
6:      **end if**
7:      $non\_zero\_ratio \leftarrow$ proportion of non-zero consumption in $group\_data$
8:      **if** $non\_zero\_ratio <$ config.sparsity_threshold **then**
9:          **Skip group**   ▷ Data too sparse
10:      **end if**
11:      $acf\_values \leftarrow$ autocorrelation function of consumption in $group\_data$
12:      **if** $\max(acf\_values[1 :]) <$ config.min_acf_threshold **then**
13:          **Skip group**   ▷ Insufficient autocorrelation
14:      **end if**
15:      **if** no start date $\geq$ config.min_start_date in $group\_data$ **then**
16:          **Skip group**   ▷ No recent data
17:      **end if**
18:      $variance \leftarrow$ variance of consumption in $group\_data$
19:      **if** $variance <$ config.min_variance_threshold **then**
20:          **Skip group**   ▷ Variance too low
21:      **end if**
22:      $missing\_ratio \leftarrow$ maximum missing ratio for features in $group\_data$
23:      **if** $missing\_ratio >$ config.max_missing_ratio **then**
24:          **Skip group**   ▷ Feature missing data too high
25:      **end if**
26:      $skewness \leftarrow$ skewness of consumption in $group\_data$
27:      **if** $|skewness| >$ config.max_skewness **then**
28:          **Skip group**   ▷ Target variable too skewed
29:      **end if**
30:      $correlation \leftarrow$ correlation of consumption with lagged features in $group\_data$
31:      **if** $|correlation| <$ config.min_correlation **then**
32:          **Skip group**   ▷ Insufficient correlation with features
33:      **end if**
34:      $final\_df \leftarrow final\_df \cup group\_data$
35: **end for**
36: **return** $final\_df$